

# Web App Testing



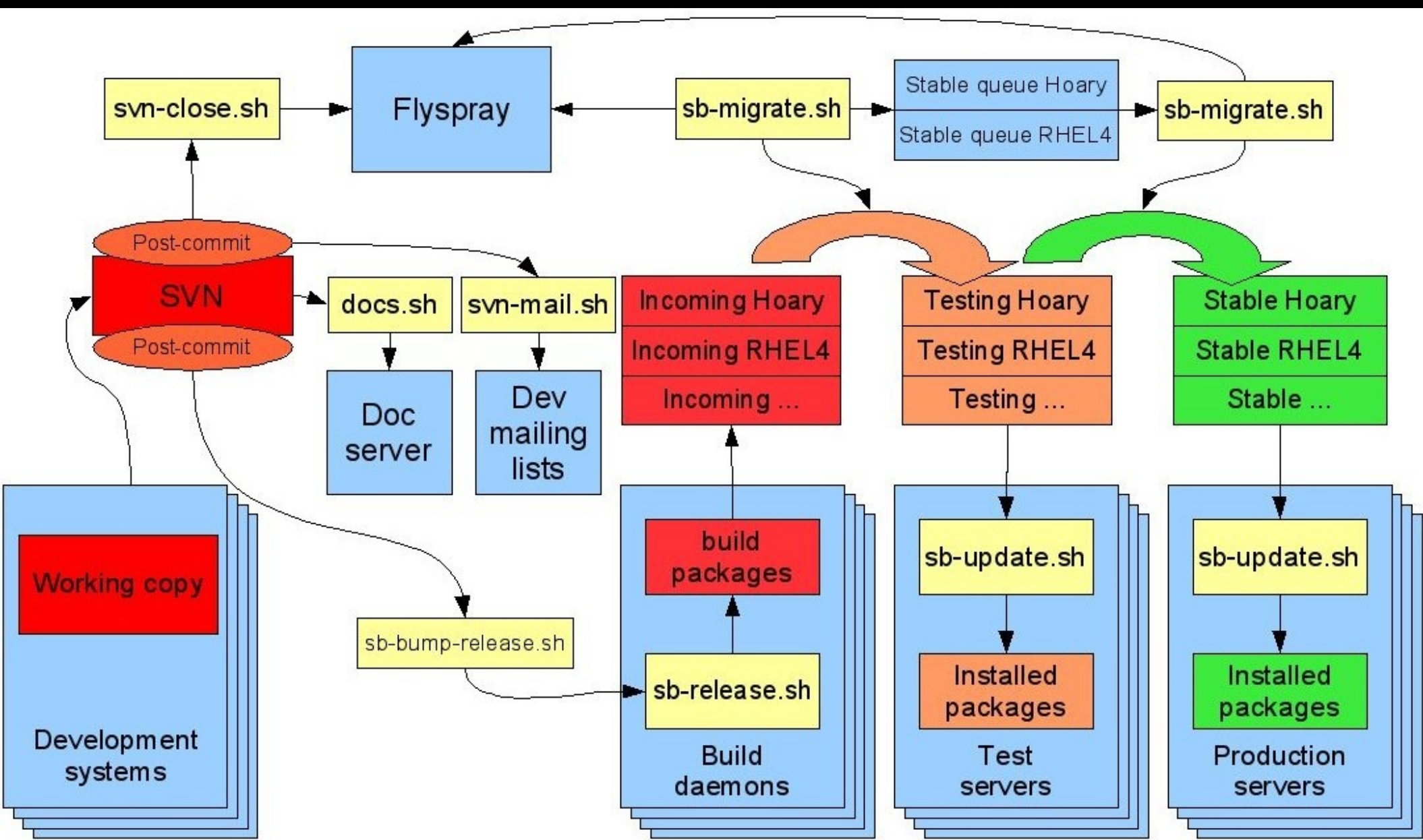
## *Simulating User / Browser Activity*

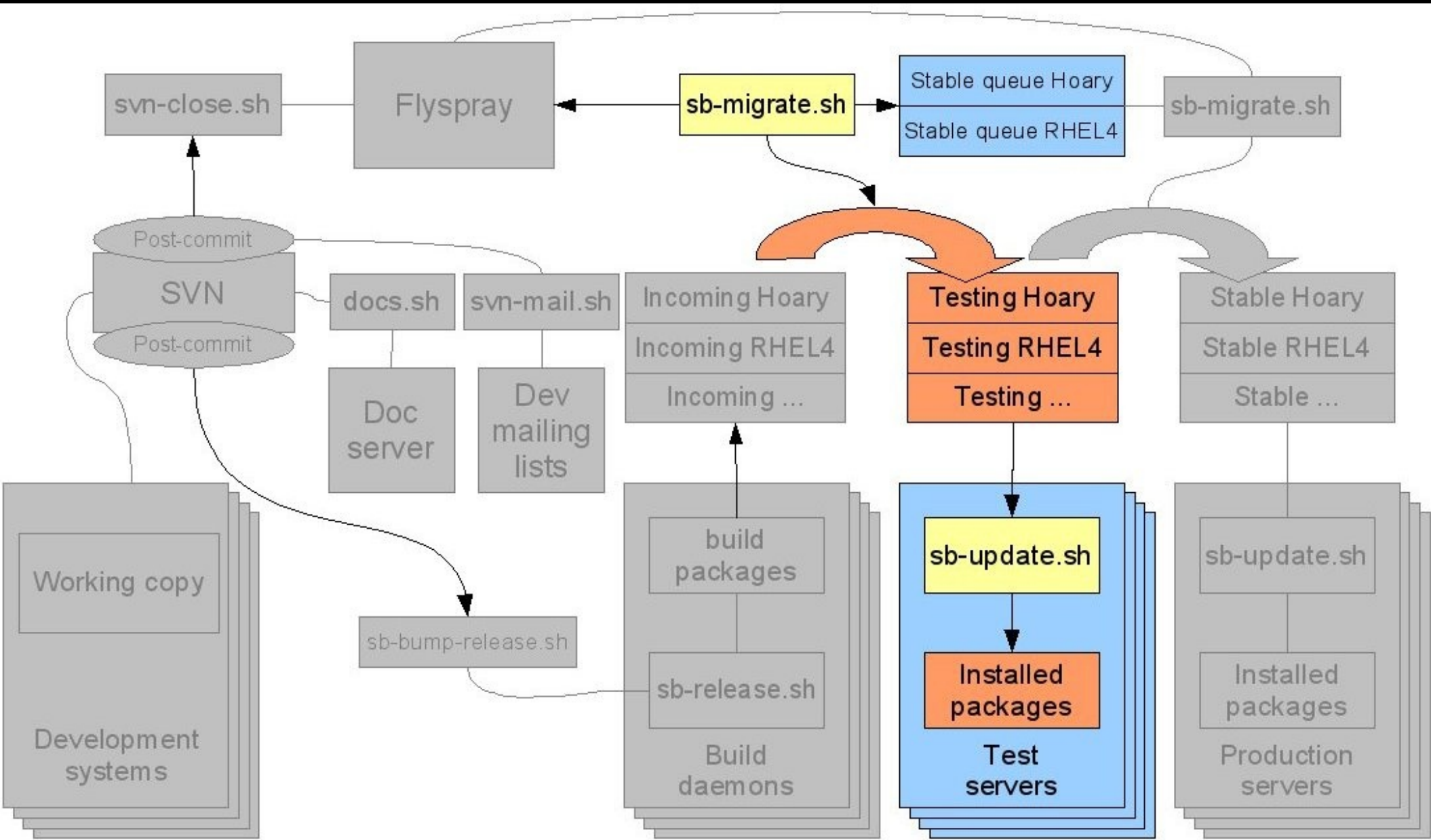
**Jonathan Oxe**

**August 28<sup>th</sup>, 2007  
IVT Tech Talk**

**Functional Tests:**  
**Treat app as a black box**

# **Unit Tests:** **Introspect app classes**





- 1: Human functional tests**
- 2: Auto functional tests**
- 3: Auto unit tests**

**Problem:  
web apps are  
hard to test**

**Solution:  
automated web  
browser actions**

# Selenium

[www.openqa.org/selenium](http://www.openqa.org/selenium)

**Written in  
Javascript, runs  
just about  
anywhere**

# **Browsers:**

**Firefox, IE, Safari, Camino,  
SeaMonkey, Mozilla, Opera,  
and Konqueror**

# **Operating Systems:**

**Linux, MacOS, and Windows**

**Selenium Core:  
installs in app**

**1. Add Selenium Core  
to the app under test  
(/selenium)**

**2. Create test suite:  
HTML file with one  
row per test**

**3. Create tests as  
HTML files with one  
row per command**

**4. Load up  
/selenium/TestRunner.html  
inside the app**

# Problems:

- ◆ **Must be installed in web root**
- ◆ **Doesn't launch browser**
- ◆ **Selenese language is limited**
- ◆ **Can't access non-browser data**
- ◆ **Limited by XSS restrictions**

**Selenium IDE:  
runs in Firefox**

**Firefox plugin to  
record Selenium  
scripts for playback  
in any browser**

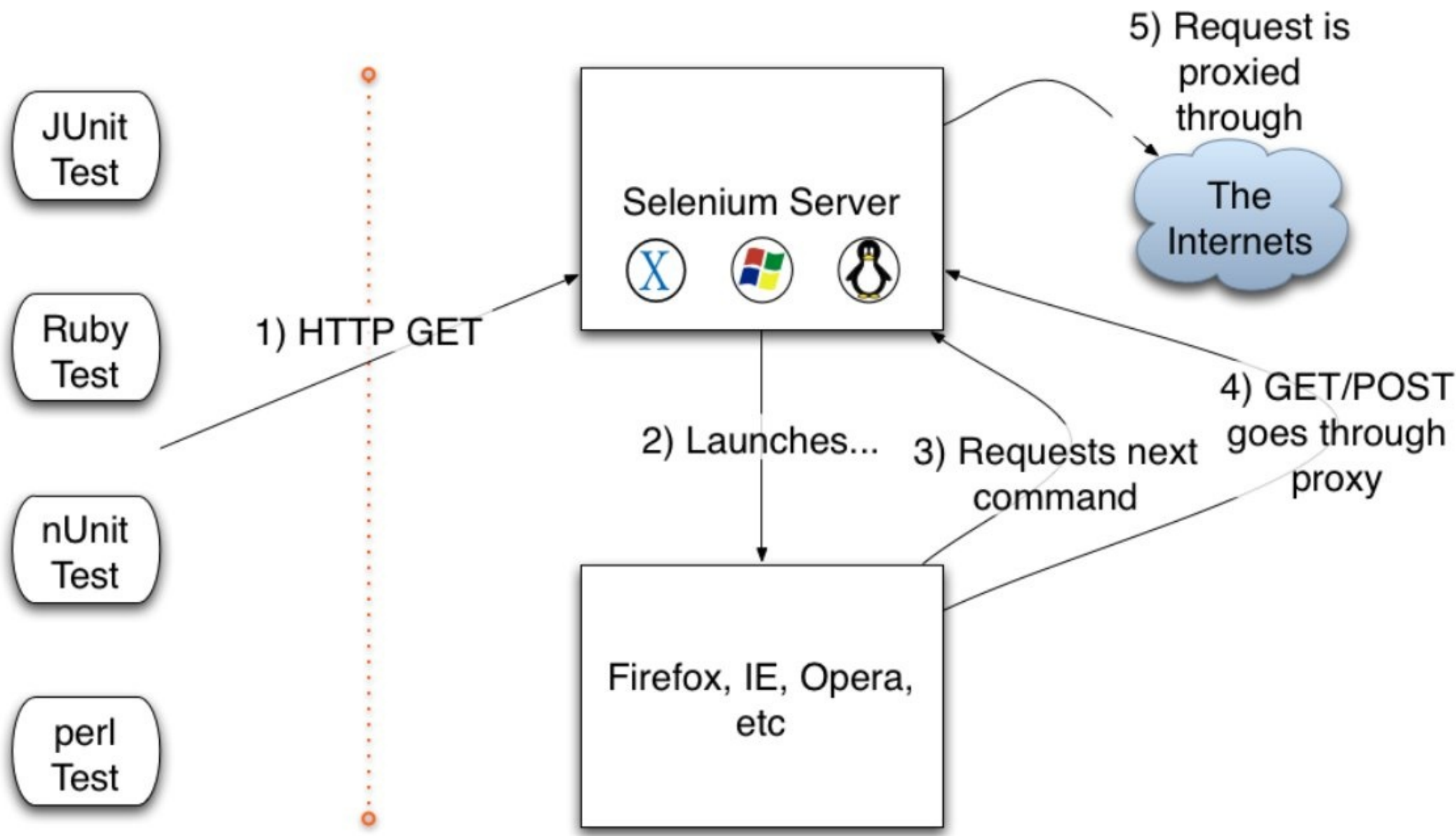
**Selenium RC:  
remote control / proxy**

**Allows complete  
automation of test  
execution**

**Write tests in any  
language**

**Doesn't require  
Selenium Core  
installed in app**

**Solves XSS and other  
security issues**



**Step 1: Create “tests”  
dir in each module**

**Step 1: Create “tests”  
dir in each module**

**Step 2: Create  
“testsuite.html” file**

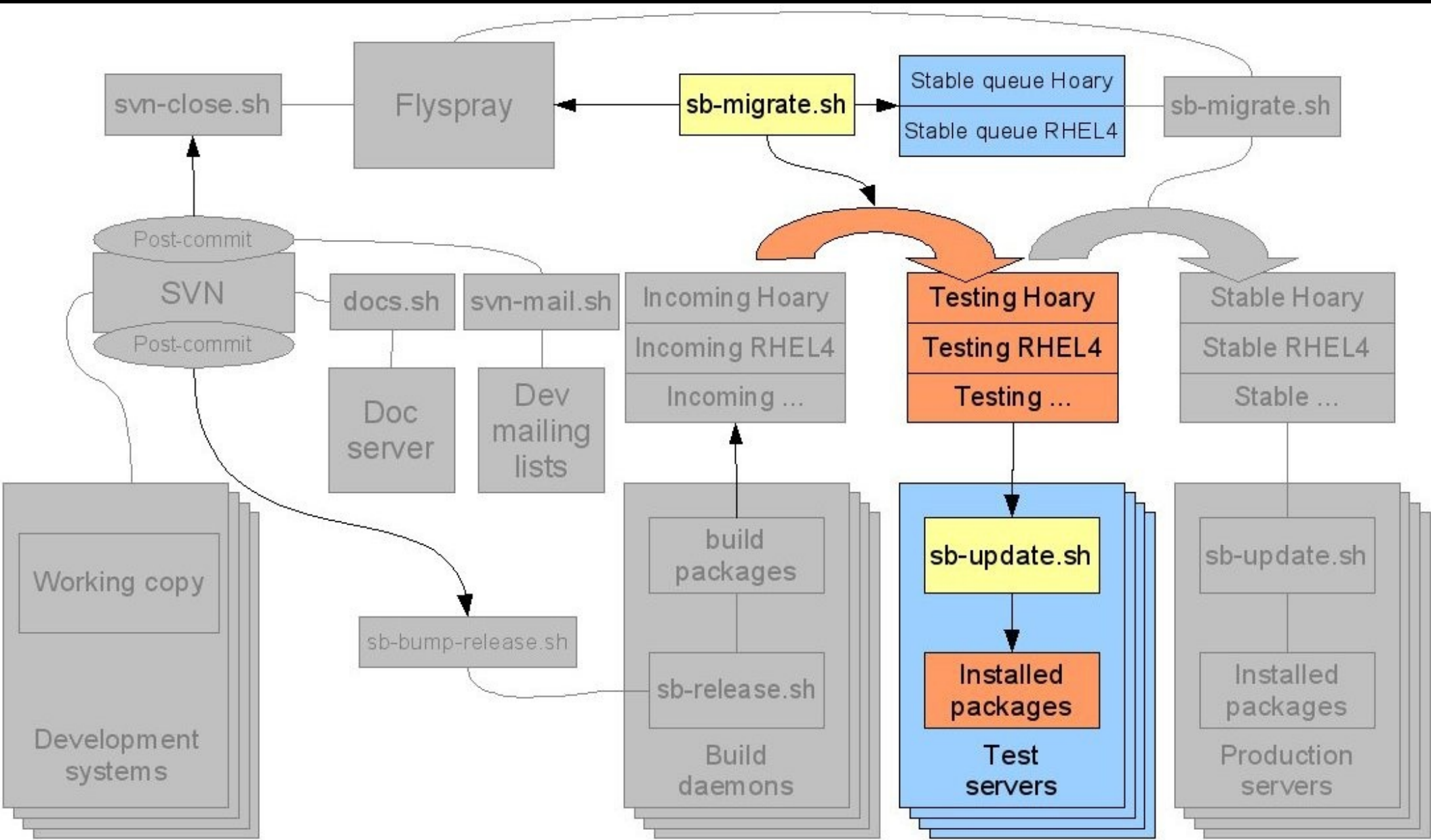
**Step 1: Create “tests”  
dir in each module**

**Step 2: Create  
“testsuite.html” file**

**Step 3: Create files  
for each test**

**A test harness will  
execute all tests  
inside a clean VM  
prior to every  
package release**

**Package release will  
depend on 100% test  
pass: results in BTS**



**What now?**

# **Install Selenium IDE in Firefox**

**Record some tests for  
your modules and  
put them in “tests”**

***Every time you fix a bug, write a test that fails on that bug first.***

***Then fix it.***

# More information



These slides are at:

[jon.oxer.com.au/talks](http://jon.oxer.com.au/talks)

Example test:

[trunk/wiki/tests/test-admin.html](http://trunk/wiki/tests/test-admin.html)

Main Selenium site:

[www.openqa.org/selenium](http://www.openqa.org/selenium)

**Thanks for listening :-)**